

AI ROADMAP & 90-DAY PLAN

KESTREL STUDIO

What to build, what to buy, what to skip — and how the first 90 days are actually run.

Sample report based on a real engagement. Client name and identifying details have been changed.

CLIENT

Kestrel Studio

European digital agency · 24 people · ~20 website projects a year

SCOPE

Decision & plan in one

Audit · build / buy / skip · impact · 90-day plan with owners

THE QUESTION

"Where does AI actually pay off for us, and what is noise?"

WORKING DETAIL

Task-level breakdowns, workflow configurations and the spec library live in your own workspace. Pages that summarise them link to the full version.

PREPARED BY

Ihor Klymchuk

Fractional CTO · September 2026

01 EXECUTIVE SUMMARY

AI pays off in two places in your studio: how you build, and how you administer projects. Everywhere else – buy a tool, or leave it alone.

VERDICT

ADOPT

Move delivery to Spec-Driven Development and automate project admin with n8n. Buy tools for everything else; skip AI where it competes with your design work.

WHAT THIS IS NOT

Not a plan to shrink the team. The same 24 people deliver more projects at a lower price, with Seniors moving into a control role instead of production typing.

THE FIVE MOVES

- 01 Seniors write block specs, Claude Code builds the blocks. Your three Seniors stop typing markup and start owning specs and review.
- 02 Three n8n automations take admin off the PMs. Notes, brief-to-spec and the weekly client update – drafted by AI, approved by a person. PM admin drops from ~6 to ~2 hours a week.
- 03 One AI stack for the whole studio. No more personal accounts.
- 04 Write the AI policy before the tools arrive. One page, signed by everyone.
- 05 Re-price standard websites once delivery is 6–8 weeks. Lower price, same margin.

IMPACT AFTER 90 DAYS

Ranges, not promises – and only if the plan runs in the order given

MEASURE	TODAY	AFTER 90 DAYS	WHAT MOVES IT
Typical project delivery	10–12 weeks	6–8 weeks	Spec-Driven Development; fewer handoff loops
Development hours per site	~300 h	170–190 h	Blocks generated from specs instead of hand-built
Price of a standard website	current list price	–20–25%	Fewer hours per project – margin per project kept

02 WHERE THE TIME GOES TODAY

Six stages, 10–12 weeks, ~300 development hours. The process is sound – handoff and development are where the hours leak.



— Sand-topped stages are where AI changes the work. Grey stages stay as they are.

WHERE THE ~300 DEVELOPMENT HOURS GO



● Repetitive build · ~165 h

Figma components turned into Gutenberg blocks and ACF fields, responsive fixes. The same shapes on every project – this is what AI takes over first.

● Rework · ~45 h

Corrections after handoff, because behaviour and edge cases were never written down. Pure waste.

● Complex logic & integrations · ~90 h

Forms, third-party systems, migrations. This stays with your developers – AI assists, people own it.

FIVE FINDINGS FROM THE AUDIT Hours per typical project unless stated

01	Requirements get lost between the call and Jira Notes live in Notion, Jira comments and notebooks. On reviewed projects, roughly one in four client decisions never reached a ticket.	10–15 h
02	Figma-to-WordPress blocks are built by hand, every time Across four recent projects, 60–70% of blocks built were structurally the same – differing only in styling and field names.	~165 h
03	Rework after design handoff Handoff is a conversation, not a document. States, long content and breakpoints get decided twice – once by the designer, once by the developer.	~45 h
04	PMs retype meeting notes and status reports ~18 h a week across three PMs on transcription and formatting – close to half a role that is not managing projects or clients.	~18 h/wk
05	No shared AI tools and no policy Personal accounts, no shared prompt library, no written rule on client material – a lost compounding effect and an open confidentiality question.	Risk

FULL DETAIL

Project-by-project evidence behind each finding, with the ticket samples – [Audit evidence pack ↗](#)

03 THE WHOLE DECISION ON ONE PAGE

Build means you set it up once and own it. Buy means you pay monthly and change nothing. Skip means it does not earn its place in your studio.

ITEM	DECISION	SETUP EFFORT	IMPACT	PAYBACK
Spec-Driven Development delivery model	BUILD	60–80 h	Highest. Cuts ~110–130 h per project and 3–4 weeks of delivery time	1st project
Shared block spec library	BUILD	20–30 h	Compounding – each project starts closer to done than the last	2nd–3rd project
Automation A: meetings → Notion → Jira	BUILD	12–16 h	~2 h/week per PM, and requirements stop disappearing	~3 weeks
Automation B: client brief → structured spec	BUILD	8–12 h	Projects start with a written spec – feeds the SDD process directly	~1 month
Automation C: weekly client status report	BUILD	6–10 h	~1.5 h/week per PM, and clients get a consistent weekly update	~1 month
AI policy and prompt library	BUILD	8–10 h	Removes the confidentiality question from client conversations	Immediate
Claude Team · whole studio	BUY	1–2 h	Shared workspace, shared prompts, data not used for training	1st month
Claude Code · 7 developers	BUY	4–6 h	The engine behind SDD – no SDD without it	1st project
Meeting notetaker · PMs and sales	BUY	2–3 h	Feeds automation A – without transcripts there is nothing to automate	1st month
n8n · automation platform	BUY	2–4 h	Hosts all three workflows; no custom code to maintain	1st month
Figma AI features	BUY	0 h	Small but free – renaming, placeholder copy, tidy-up tasks	n/a

SKIP – AND WHY Revisit in twelve months, not sooner

AI-generated final designs

Competes with the thing clients choose you for. AI stays on housekeeping inside the design stage.

Chatbots on client sites by default

Low demand, long support tail. Build one only when a client asks and pays for it as separate scope.

Fine-tuning your own model

Twenty projects a year is not a training set. Written specs and a prompt library give the same consistency.

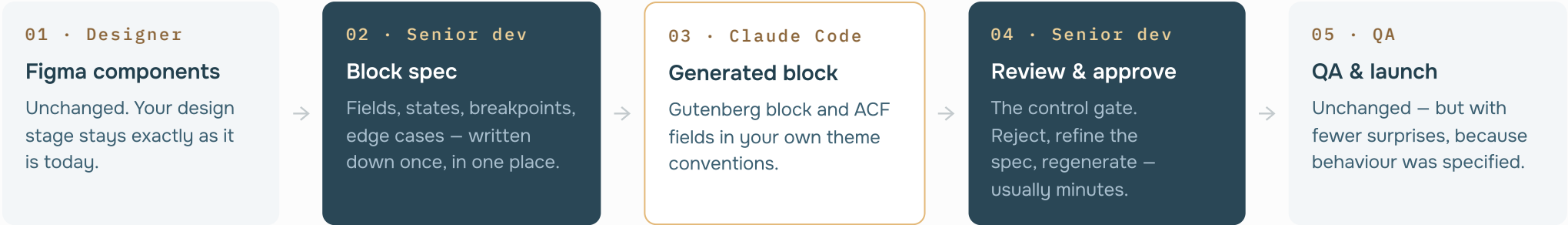
AI site builders replacing WordPress

Breaks a stack that works and a handover clients accept – a migration problem you would own for years.

04 · Build 01

SPEC-DRIVEN DEVELOPMENT

Your Senior developers describe what a block must do. An AI agent writes it. The Senior approves it. Nothing ships unreviewed.



WHAT EACH ROLE DOES INSTEAD

ROLE	TODAY	UNDER SDD
3 Senior developers	Build the hard parts, review pull requests when there is time	Own the specs and the review gate – their judgement reaches every block
4 mid-level developers	Most of the repetitive block building and responsive fixes	Integrations, complex logic, migrations, and specs for simpler blocks
8 designers	Design, then re-explain it during development	Design, then answer spec questions once, in writing, before build starts
3 project managers	Chase status across three tools	Track against specs – progress becomes countable instead of reported

EXPECTED EFFECT PER TYPICAL PROJECT

-110-130 h

Development hours, mostly out of the repetitive-build 55%

-3-4 weeks

Calendar time, from fewer handoff and rework loops

Two conditions. A spec is never optional, and review is a named step – code no Senior approved does not reach QA, whatever the deadline looks like.

THE PILOT CAVEAT

The first pilot will feel slower, not faster – specs are being written for the first time. Judge the model on the second and third project, and tell the team that up front.

05 WHAT A SPEC CONTAINS, AND WHAT GETS REUSED

A full spec takes a Senior 45–90 minutes and replaces about a day of back-and-forth. Ten of them cover almost every project you sell.

BLOCK SPEC TEMPLATE · 12 SECTIONS

01 Name & purpose	07 Content limits
02 Where used	08 Accessibility
03 Fields	09 Editor experience
04 Variants	10 Performance notes
05 States	11 Out of scope
06 Breakpoints	12 Acceptance & reviewer

The four in bold are where rework comes from. The Senior writes 01–06 and 09–12; the designer answers 05, 06 and 07 in writing, once, before build.

LIBRARY V1 · TEN ENTRIES ~15 h for all ten

BLOCK	SPEC	WAVE
Hero with media	1.5 h	FIRST
Text + image split	1 h	FIRST
Card grid	2 h	FIRST
Accordion / FAQ	1.5 h	FIRST
Contact form	2.5 h	FIRST
CTA band	0.5 h	FIRST
Case study grid	2 h	SECOND
Testimonial slider	1.5 h	SECOND
Stats band	1 h	SECOND
Team / people grid	1.5 h	SECOND

Write the six First entries during the pilots; add the rest as projects need them, and delete anything unused for three projects.

DEFINITION OF DONE · PER BLOCK

- A spec exists and is linked on the ticket
- A Senior has ticked the acceptance checks
- All named states render, including empty and error
- Editor view is sane: fields grouped, help text present
- Spec updated if the build changed anything

SENIOR REVIEW · 10–20 MIN PER BLOCK

- Does it match the spec, or only look like the design?
- Theme conventions: naming, placement, ACF JSON committed
- No new dependency or plugin added quietly
- Reject by fixing the spec, not by patching the code

FULL DETAIL

The template with every section filled, a worked example and the Claude Code conventions file – [Spec template & library ↗](#)

06 • Build 02 PROJECT ADMIN ON N8N

Three workflows, no custom code. Every one ends with a person approving a draft – the automation never speaks to a client or creates a ticket on its own.

Workflow A

Meetings → Notion → Jira

Fixes finding 01 – decisions no longer live in one person's memory.

Trigger Notetaker posts a finished call transcript

Steps Claude extracts decisions, action items, owners and open questions → a meeting page is created in the project's Notion space → draft Jira tickets are prepared, not created

Output **PM approves** → tickets land in Jira with the Notion page linked

~2 h/wk

saved per PM

12–16 h

to set up

Workflow B

Client brief → structured spec

Feeds Spec-Driven Development – a project starts written down.

Trigger Client submits the intake form

Steps Claude turns free-text answers into a structured draft – pages, blocks, integrations, content needs – lists gaps and contradictions as explicit open questions, and writes it to Notion in the studio's template

Output **PM and Senior refine** → becomes the project's base spec

4–6 h

saved per new project

8–12 h

to set up

Workflow C

Weekly client status report

Fixes finding 04 – and gives every client the same rhythm.

Trigger Schedule – every Thursday morning, per active project

Steps n8n pulls the week's Jira movement and open blockers → Claude drafts the update in plain client language (done, next, needs you) → the draft lands in the PM's queue, never in the client's inbox

Output **PM edits and sends** – typically 10 minutes instead of an hour

~1.5 h/wk

saved per PM

6–10 h

to set up

FULL DETAIL Node-by-node configuration for all three workflows – prompts, field mappings, approval queues and error paths – [Workflow designs ↗](#)

07 THE FIRST 90 DAYS ON ONE PAGE

Six workstreams, run inside live projects. Sand and dark bars are active work; pale bars are where a workstream runs but needs no new effort. Nothing here requires stopping delivery or hiring anyone.

	DAYS 1–30 · FOUNDATIONS	DAYS 31–60 · PROVE IT TWICE	DAYS 61–90 · MAKE IT DEFAULT
Policy & tools	Policy signed · Claude Team live	Prompt library grows	Tool review
Automation A	Build & run — meetings → Jira	In daily use	In daily use
Automations B & C		Build — brief → spec, weekly report	In daily use
SDD pilots	Pilot 1 · Seniors only	Pilot 2 · mid-level devs join	Default for new projects
Spec library	First specs written on pilot 1	Library v1 — 10 entries	Grows per project
Metrics & pricing	Baseline from 3 past projects	Pilot 1 measured	New pricing live

GATE · DAY 30

Automation A running, first specs written

Policy signed, personal accounts closed, notetaker on every call.

GATE · DAY 60

Two pilots delivered and measured

If pilot 1 did not beat the baseline on two of the four metrics, run a third pilot instead of rolling out.

GATE · DAY 90

New price live, SDD the default

Spec and review in the definition of done; next automation chosen from what PMs still do by hand.

08

WHO OWNS WHAT

27 tasks · 128–168 h

Six roles, named by function. Fill in the actual names before day one – an unnamed owner is the most common reason this kind of plan stalls.

ROLE	WHO	ACCOUNTABLE FOR	TIME IN 90 DAYS
Sponsor	Owner / leadership	Policy signed, budget approved, pricing changed at day 90. Removes blockers, runs no tasks.	10–14 h
SDD lead	Senior dev 1	Spec template, Claude Code conventions, the review gate – the most important role here.	40–52 h
Spec library owner	Senior dev 2	Turning pilot specs into reusable library entries, and keeping them current.	18–24 h
n8n owner	PM 1 or Senior dev 3	All three workflows: building them, watching failed runs, fixing them.	34–44 h
Delivery PM	PM 2	The two pilot projects, the approval queues, and the four metrics – baseline and monthly.	22–28 h
Tools admin	Ops / office manager	Seats, billing, access – and closing the personal accounts in week one.	6–8 h

WORK BREAKDOWN · WHAT EACH PHASE ACTUALLY CONTAINS

PHASE	TASKS	EFFORT	THE HEAVY ITEMS
Days 1–30	11 tasks	52–68 h	Policy and seats, notetaker chosen, workflow A built, Claude Code conventions, first six specs, baseline from three past projects
Days 31–60	9 tasks	46–60 h	Intake form rebuilt, workflows B and C, pilot 2 with mid-level developers writing specs, library v1, review checklist, pilot 1 measured
Days 61–90	7 tasks	30–40 h	Spec and review written into Jira states, all seven developers trained, new price list, proposals updated, tool review, next automation chosen

FULL DETAIL

All 27 tasks numbered, with owner, effort, week and a written "done when" for each – [Full work breakdown ↗](#)

CADENCE

Stand-up weekly, 30 min

Gate review day 30 / 60 / 90, 60 min

Retro after each pilot, 45 min

09 WHAT IT COSTS, AND WHEN IT PAYS BACK

Five lines on the invoice. Setup effort is your own team's time, not a bill – and both are small against the hours recovered.

TOOL	PURPOSE	~MONTHLY
Claude Team ~24 seats · everyone	One shared workspace instead of personal accounts. Shared prompts, business plan with no training on your data, an admin who can answer a client's security question.	€600–720
Claude Code 7 developer seats	The engine behind SDD: generates Gutenberg blocks and ACF fields from written specs inside your own theme. Usage-based – heavy build weeks cost more.	€700–1,400
Meeting notetaker Granola or Fathom · 5–6 seats	Produces the transcripts workflow A depends on. Pick on how it handles your languages, then standardise on one.	€100–160
n8n cloud, one named owner	Hosts all three workflows. Start on cloud – no server to maintain. Self-host later only if a client contract requires it.	€50–120
Figma AI features existing plan · 8 designers	Layer renaming, placeholder content, quick variants. Housekeeping inside the design stage – not a design generator.	included
TOTAL	Whole studio, all tools, no custom software to maintain	€1,450–2,400

THE TRADE, IN ROUND NUMBERS

One-time setup effort, all items	120–170 h
Monthly tool cost, whole studio	€1,450–2,400
Hours recovered per typical project	110–130 h
Admin hours recovered across three PMs	~12 h/week

PAYBACK

2–3 months

With roughly 20 projects a year, the hours recovered on two or three projects already cover a full year of tools and all of the setup effort.

No new hires, no custom software, no infrastructure project – subscriptions you can cancel and a process your own people own.

Buy in this order. Claude Team and the notetaker first – no engineering needed; Claude Code and n8n when the pilot starts.

One owner per tool. Unowned tools become unused subscriptions within two months – and drop anything without a visible user at the day-90 review.

10 HOW YOU WILL KNOW IT WORKED

Four numbers, filled in once a month. Fill the baseline column in week three – without it, nothing in this plan can be proved or disproved at day 90.

METRIC	WHERE IT COMES FROM	BASELINE	PILOT 1	PILOT 2	DAY-90 TARGET
Delivery time	Project start and launch dates	10–12 wks	–	–	6–8 wks
Development hours	Time tracking, dev tickets only	~300 h	–	–	170–190 h
Rework after handoff	Tickets labelled rework	~45 h	–	–	under 20 h
PM admin hours	One self-reported line per PM per week	~6 h/wk	–	–	~2 h/wk

— The monthly ritual is 30 minutes: four numbers into the sheet, one line on what moved and why, one decision – keep, adjust or stop – recorded rather than implied.

KEY RISKS, HOW EACH IS HELD DOWN, AND WHAT ROLLBACK LOOKS LIKE

● Quality

Quality drift if specs aren't reviewed

No generated block ships without a Senior review – it is in the definition of done, not a matter of goodwill.
If it slips: stop generating on that project and hand-build the rest; keep the specs.

● Client data

Client material in AI tools

Briefs and business context only – never customer databases or credentials. Paid plans with no training on your data, plus the signed one-page policy overleaf.

● Adoption

The team doesn't take it up

Two pilots with volunteers first; nothing is mandatory until it has worked twice. **If it stalls:** do not mandate – pair the developer it works for with the rest and move the gate a month.

● Noise

Automation clutters Jira

Automations create drafts, not tickets; unapproved drafts expire. **If one misfires:** disable that workflow in n8n – the manual process is unchanged and nothing in delivery depends on it.

WHERE THIS PLAN ENDS

Day 90 – the studio owns it

Everything here is run by your own people on tools you pay for monthly: the specs, the workflows and the policy live in your Notion, your repos and your n8n. Quarter two is a narrower question – measure first, then decide what is worth automating next.

11 AI POLICY — READY TO SIGN

One page, plain language, no legal padding. Replace the bracketed parts, then circulate it in week one — it is task 1.1 of the plan.

ALLOWED

Briefs, public information, your own drafts, code from our repositories, and anonymised client business context — in the studio's paid Claude workspace only.

Generated output is a draft. A named person is responsible for anything that reaches a client.

HOW WE WORK

We use AI to move faster on production work — specs, code, notes, drafts. We do not sell generated design as our design work. Every client-facing deliverable is reviewed by the person whose name is on it.

If a client asks, we tell them plainly: which tools, on what material, with what review.

CLIENT-FACING VERSION

One paragraph that sales can use unchanged: we use AI on production work under written specs and human review; your confidential material stays out of consumer tools; the people on your project are the same people who sign off on it.

NOT ALLOWED

Personal AI accounts for client work. Customer databases, personal data or production credentials in any prompt. Material a client has marked confidential, unless they have agreed in writing. Free or consumer AI tools where the provider may train on the input.

EXCEPTIONS & QUESTIONS

Anything outside this page is decided by **[name, role]** before it happens, not after. Asking is never the wrong move; assuming is.

ACKNOWLEDGEMENT

Name and role

Date · version **[v1]**, reviewed every six months